IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Patent Application of:
Philip Rousselle

Application No.: 10/672,795

Filed: September 26, 2003

For: Implementing request/reply programming semantics using publish/subscribe middleware

Group Art Unit: 2192

Conf. No.: 9466

Examiner: Michael J. Yigdall

Mail Stop Appeal Brief - Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

APPEAL BRIEF
UNDER 37 C.F.R. § 41.37

Dear Sir:

This is an appeal to the Board of Patent Appeal and Interferences from the Decision of the Examiner finally rejecting claims 1-25. This brief is in furtherance to the Notice of Appeal filed on May 27, 2008. Provision for the payment of the filing fee is enclosed herewith.

TABLE OF CONTENTS

I. REAL PARTY IN INTEREST 3
II. RELATED APPEALS AND INTERFERENCES 3
III. STATUS OF CLAIMS 3
IV. STATUS OF AMENDMENTS 3
V. SUMMARY OF CLAIMED SUBJECT MATTER 3
A. Independent Claim 1 3
B. Independent Claim 19 3
C. Independent Claim 24 4
VI. GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL 4
VII. ARGUMENT 4
A. The Codella reference 4
B. Traversal of the rejection 5
VIII. CLAIMS APPENDIX 11
IX. EVIDENCE APPENDIX 16
X. RELATED PROCEEDINGS APPENDIX 17

I. REAL PARTY IN INTEREST

Inceptia LLC, a Delaware limited liability company, is the real party in interest for all issues related to this application.

II. RELATED APPEALS AND INTERFERENCES

None.

III. STATUS OF CLAIMS

Claims 1 – 25 are cancelled.

Claims 1 – 25 stand rejected and are the subject of this appeal.

IV. STATUS OF AMENDMENTS

On May 27, 2008, concurrent with the filing of the Notice of Appeal, an amendment was filed amending certain of the claims. The Examiner did not accept the amendment and the amendment was therefore not entered. No other amendment, other than the aforementioned was filed after the final Office Action (hereinafter "Office Action") mailed on November 27, 2007.

V. SUMMARY OF CLAIMED SUBJECT MATTER

A. Independent Claim 1

The subject matter of independent claim 1 is directed to a method for facilitating communications between components of a distributed application comprising the steps of: receiving a request from a first distributed application component, wherein a recipient of said request is a second distributed application component; and publishing said request on a first publish/subscribe request topic, wherein said first publish/subscribe topic is identified by a first property of said second distributed application component.

B. Independent Claim 19

The subject matter of independent claim 19 is directed to a system for facilitating request/reply communications among components of a distributed application comprising a distributed computing system, tangibly embodying thereon: a publish/subscribe request topic for every type of distributed application component; a publish/subscribe reply topic for every

type of distributed application component; and for each distributed application component, a publisher on every publish/subscribe request topic within a portion of said publish/subscribe request topics; a publisher on every publish/subscribe reply topic within a portion of said publish/subscribe reply topics; a subscription on the publish/subscribe request topic pertaining to a type of said distributed application component; and a subscription on the publish/subscribe reply topic pertaining to a type of said distributed application component.

C. Independent Claim 24

The subject matter of Claim 24 is directed to a method of communicating messages between components of a distributed application comprising the steps of: receiving a message formulated according to request/reply semantics from a first distributed application component; translating said message into publish/subscribe communications implemented by a publish/subscribe middleware program; and forwarding said translated message to a second distributed application component.

VI. GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

1. Whether claims 1-25 are unpatentable pursuant to 35 U.S.C. § 102 as being anticipated by U.S. Patent No. 6,804,818 to Codella et al. (hereinafter "Codella").

VII. ARGUMENT

Claims 1-25 were rejected pursuant to 35 U.S.C. § 102(e), as being anticipated by Codella. Applicant respectfully traverses the Examiner's rejection, and respectfully submits that none of the pending claims are anticipated by Codella. Codella fails to disclose either explicitly or inherently "each and every limitation" of the pending claims – the fundamental requirement for an anticipation rejection.

A. The Codella reference

Codella discloses mechanisms for sending and receiving **anonymous** invocations by integrating software based on Object-Oriented Middleware with software based on Message-

Oriented Middleware. One object of Codella's invention is to permit an object-oriented component, (i.e. a message bean), to perform anonymous invocations that are serviced by other message beans or by message-oriented servers in such a way that the requesting message bean is unaware of whether the server of the anonymous invocation is either a message bean or a message-oriented server. Thus, the anonymous requests taught by Codella do not identify the recipient-message bean or component. By clear contrast, the invention as embodied by the pending claims, do not involve anonymous invocations. The recipient of a request generated by a component of the present invention is not anonymous – but indeed identifiable.

B. Traversal of the rejection

Claims 1 & 18

Codella does not teach at least the limitation: *"wherein said first publish/subscribe topic is identified by a first property of said second distributed application component."* The Office Action cites Codella at col. 19, lines 37-54, as allegedly disclosing this limitation. However, that portion of Codella merely discloses the deployment descriptor properties for the input ports of a message bean, but does not disclose how to identify a publish/subscribe topic for publishing a request originating from a different message bean.

Moreover, Claim 1 when read in light of the specification requires that the first publish/subscribe topic is identified using and based on a first property of the second component. In other words, an attribute, type or characteristic (i.e. a property) of the second component is used for identifying the first publish/subscribe topic.

The term "property" is defined by the *American Heritage Dictionaries* as:

a. **A characteristic trait or peculiarity, especially one serving to define or describe its possessor.**
b. **A characteristic attribute possessed by all members of a class**

See: http://www.answers.com/property

The Examiner in the *Response to Arguments* section of the Office Action, at p. 3, notes that Codella at 7:3-39 describes an input port (i.e. a component) *property* that identifies a destination – that is a public/subscribe topic. However, the term "property" isn't mentioned in that portion of Codella or anywhere else in Codella. Rather, the cited portion of Codella merely lists "definitions" which can be specified by an input port. The Examiner, in rejecting the claims, apparently equates "definition" with "property".

However, the term "property" is to be construed by its ordinary meaning and in light of the specification. To the extent, the Examiner understands the term "property" to mean "definition" or some other form of "data field", it is clear that the term "property" as used in the pending claims means the characteristics or type **of** the second component, as opposed to a data field or definition included **within** the second component.

By applying a different meaning to the term "property" the Examiner has taken the passage *"wherein said first publish/subscribe topic is identified by a first property of said second distributed application component"*, out of context.

As an analogy: Suppose one were to say "I identified a book in the library based on the property of a book I have at home". The plain meaning of this sentence is that the person selected a book in the library using a property/characteristic of a book he had at home. Thus, he/she may select a science fiction book by using the "science fiction" property/characteristic of the book he had at home. But using the Examiner's definition of property, this sentence would suddenly acquire an entirely different meaning. It would essentially mean that somewhere within the book at home (i.e.a data entry within the book) there is disclosed the identity of a book in the library.

Clearly, when reading the claims as a whole and in light of the specification, and when deferring to ordinary dictionary meanings, it is readily recognizable that the invention of Codella is not a reference relevant to the invention disclosed in Claim 1 and Claim 18.

Claim 2

The traversal of claim 1 is hereby incorporated. Furthermore, Codella does not disclose at least the limitation: "*wherein said first property is a type of said second distributed application component*". The Office Action cites Codella at col. 19, lines 37-54, as allegedly disclosing this limitation. However, that portion of Codella merely discloses the deployment descriptor properties for the input ports of a message bean, but provide no disclosure for identifying a publish/subscribe topic based on a type of message bean.

The Examiner in the *Response to Arguments* section of the Office Action, at p. 3, refers to Codella at 19:5-25 as disclosing a "type" of message bean. That portion of Codella disclosed descriptors and not "types". It is unclear what the Examiner means by referring to the "Messageport" or "Messageinputport" as "types".

Claim 3

The traversal of claim 2 is hereby incorporated. Additionally, Codella does not disclose the limitation: "*wherein said recipient is identified by a second property of said second distributed application component included within said request*". The Office Action cites Codella at col. 11, lines 6-11, as allegedly disclosing this limitation. However, nothing in that portion of Codella suggests that an identification property of the second component is included within the request generated by the first component. In fact, including an identification property of the recipient component in the request is contradictory to Codella's objective of providing an anonymous messaging system.

The Examiner in the *Response to Arguments* section of the Office Action, at p. 4, notes that a "field" as disclosed in Codella is a "property". As noted, the term "property" as used herein is based on its ordinary meaning and as used in light of the specification.

Claim 4

The traversal of claim 3 is hereby incorporated. Furthermore, Codella does not disclose the limitation: "*wherein said second property is a unique identifier of said second distributed application component*". The Office Action cites Codella at col. 13, lines 13-17, as allegedly

disclosing this limitation. However, unlike claim 4, that portion of Codella does not disclose that a unique identifier is included within the request generated by the message bean.

Claim 5

The traversal of claim 2 is hereby incorporated. Furthermore, Codella does not disclose at least the limitation: "*wherein said first publish/subscribe reply topic is identified by a type of said first distributed application component*". The Office Action cites Codella at col. 19, lines 37-54, as allegedly disclosing this limitation. However, that portion of Codella merely discloses the deployment descriptor properties for the input ports of a message bean, but does not provide any disclosure for how to identify reply topics for subscribing a given message bean – let alone basing the identification on component type.

Claim 7

The traversal of claim 1 is hereby incorporated. Furthermore, Codella does not disclose at least the limitations: (a) "*wherein said second publish/subscribe request topic is identified by a type of said first distributed application component*"; (b) "*wherein said second publish/subscribe reply topic is identified by a type of said third distributed application component*". The Office Action cites Codella at col. 19, lines 37-54, as allegedly disclosing these limitations. However, that portion of Codella merely discloses the deployment descriptor properties for the input ports of a message bean, but provides no disclosure for how to identify a publish/subscribe topic for which to subscribe a given message bean – let alone using component types.

Claim 11-14

The traversal of claim 1 is hereby incorporated. Furthermore, Codella does not disclose the limitations: "*receiving a type of said first distributed application component, a name of said first distributed application component, a list of all other types of distributed application components that will send request or replies to said first distributed application component, and a list of all other types of distributed application components that will be receiving requests or replies from said first distributed application component*". The Office Action cites Codella at col. 7, lines 30-67, as allegedly disclosing these limitations. However, that portion of

Codella does not disclose that the type of any application component is specified while registering the first component. The deployment descriptor in Codella merely specifies the name of the input and output ports of the message bean and the names of the message bean's method and the message proxy's method, but does not specify the type of any message bean.

Claim 15-17

The traversal of claim 15, the traversal of claim 11 is hereby incorporated. Furthermore, Codella does not disclose creating a publisher or a subscription to a publish/subscribe topic based on a type of component or message bean. None of the portions of Codella cited in the Office Action discloses topics based on component types.

Claims 19-23

Codella does not disclose Codella at least publish/subscribe topics that are based on component types. The Office Action cites Codella at col. 7 lines 1-12, as allegedly having this disclosure. However, that portion of Codella merely deals with attributes of a "message bean listener" – not with attributes of a "publish/subscribe topic".

Claims 24-25

Codella does not disclose: "*translating a message formulated according to requests/reply semantics into publish/subscribe communications*". The Office Action cites col. 15 lines 61-63, for showing that a JMS destination can be a publish/subscribe topic. However, nothing in that portion (or any other portion in Codella) suggests either explicitly or by inherence, that a message that was originally formulated according to request/reply semantics can be translated to publish/subscribe communications. Moreover, Codella is devoid of any disclosure that would enable such translation. By contrast the translation of claims 24-25 are enabled according to novel implementations disclosed in the present specification.

CONCLUSION

In view of the foregoing, it is respectfully requested that the rejections be reversed.

Dated: November 28, 2008

Respectfully submitted,

Signature: /Benzion A. Wachsman/

Benzion A. Wachsman
General Manager
Inceptia LLC
Tel.: (718) 928-2213
Fax: (718) 504-9671

CLAIMS APPENDIX

1. A method for facilitating communications between components of a distributed application comprising the steps of:

receiving a request from a first distributed application component, wherein a recipient of said request is a second distributed application component; and

publishing said request on a first publish/subscribe request topic, wherein said first publish/subscribe topic is identified by a first property of said second distributed application component.

2. The method of claim 1, wherein said first property is a type of said second distributed application component.

3. The method of claim 2, wherein said recipient is identified by a second property of said second distributed application component included within said request.

4. The method of claim 3, wherein said second property is a unique identifier of said second distributed application component.

5. The method of claim 2, further comprising the steps of:

subscribing to a first publish/subscribe reply topic, wherein said first publish/subscribe reply topic is identified by a type of said first distributed application component;

forwarding a reply posted on said first publish/subscribe reply topic to said first distributed application component.

6. The method of claim 5, wherein said reply is generated by said second distributed application component in response to said request.

7. The method of claim 1, further comprising the steps of:

subscribing to a second publish/subscribe request topic, wherein said second publish/subscribe request topic is identified by a type of said first distributed application component;

forwarding a request posted on said second publish/subscribe request topic to said first distributed application component, wherein said request is generated by a third distributed application component;

receiving a reply from said first distributed application component, wherein a recipient of said reply is said third distributed application component; and

publishing said reply on a second publish/subscribe reply topic, wherein said second publish/subscribe reply topic is identified by a type of said third distributed application component.

8. The method of claim 7, wherein said second and third distributed application components are the same distributed application component.

9. The method of claim 7, further comprising the step of, prior to forwarding said request posted on said second publish/subscribe request topic,

identifying that a recipient of said request posted on said second publish/subscribe request topic is either said first distributed application component or all distributed application components.

10. The method of claim 7, further comprising the step of,

sending a callback object to said first distributed application component with said request posted on said second publish/subscribe request topic.

11. The method of claim 1, further comprising the step of,

registering said first distributed application component prior to receiving said request, wherein said step of registering comprises:

receiving a type of said first distributed application component, a name of said first distributed application component, a list of all other types of distributed application components that will send request or replies to said first distributed application component, and

a list of all other types of distributed application components that will be receiving requests or replies from said first distributed application component.

12. The method of claim 11, wherein said step of registering further comprises:

receiving a callback object, wherein said callback object directs requests from other distributed application components to said first distributed application component.

13. The method of claim 12, further comprising the step of,

invoking said callback object to deliver said request to said first distributed application component.

14. The method of claim 11, wherein said step of registering further comprises:

sending a callback object to said first distributed application component.

15. The method of claim 11, wherein said step of registering further comprises:

creating a publisher on a publish/subscribe request topic of each of said other type of distributed application component receiving a request from said first distributed application component;

creating a publisher on a publish/subscribe reply topic of each of said other type of distributed application component types receiving a reply from said first distributed application component;

creating a subscription on a publish/subscribe request topic of said type of said first distributed application component; and

creating a subscription on a publish/subscribe reply topic of said type of said first distributed application component.

16. The method of claim 15, wherein said subscription on a publish/subscribe request topic of said type of said first distributed application component includes a filter that only accepts requests addressed to said first distributed application component or all distributed application components.

17. The method of claim 15, wherein said subscription on a publish/subscribe reply topic of said type of said first distributed application component includes a filter that only accepts replies addressed to said first distributed application component.

18. The method of claim 1, wherein said request comprises one or more instructions directed toward a task to be performed by said second distributed application component.

19. A system for facilitating request/reply communications among components of a distributed application comprising a distributed computing system, tangibly embodying thereon:

a publish/subscribe request topic for every type of distributed application component;

a publish/subscribe reply topic for every type of distributed application component; and for each distributed application component,

a publisher on every publish/subscribe request topic within a portion of said publish/subscribe request topics;

a publisher on every publish/subscribe reply topic within a portion of said publish/subscribe reply topics;

a subscription on the publish/subscribe request topic pertaining to a type of said distributed application component; and

a subscription on the publish/subscribe reply topic pertaining to a type of said distributed application component.

20. The system of claim 19, wherein said portion of said publish/subscribe request topics includes publish/subscribe request topics pertaining to all types of distributed application components that receive requests from said distributed application component.

21. The system of claim 19, wherein said portion of said publish/subscribe reply topics includes publish/subscribe reply topics pertaining to all types of distributed application components that receive replies from said distributed application component.

22. The system of claim 19, further comprising:

one or more callback objects to facilitate delivery of requests and replies between said distributed application components and said publishers or subscriptions.

23. The system of claim 22, further comprising:

routing logic to route a request or reply to a particular callback object.

24. A method of communicating messages between components of a distributed application comprising the steps of:

receiving a message formulated according to request/reply semantics from a first distributed application component;

translating said message into publish/subscribe communications implemented by a publish/subscribe middleware program; and

forwarding said translated message to a second distributed application component.

25. The method of claim 24, wherein said message is a request or reply.

EVIDENCE APPENDIX

None.

RELATED PROCEEDINGS APPENDIX

None.

PTO/SB/96 (11-08)
Approved for use through 12/31/2008. OMB 0651-0031
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE

STATEMENT UNDER 37 CFR 3.73(b)

Applicant/Patent Owner: Philip Rousselle

Application No./Patent No.: 10/672,795 Filed/Issue Date: September 26, 2003

Entitled: Implementing request/reply programming semantics using publish/subscribe middleware

Inceptia LLC (Name of Assignee), a Delaware limited liability company (Type of Assignee, e.g., corporation, partnership, university, government agency, etc.)

states that it is:

1. [✓] the assignee of the entire right, title, and interest; or
2. [ ] an assignee of less than the entire right, title and interest (The extent (by percentage) of its ownership interest is _____ %)

in the patent application/patent identified above by virtue of either:

A. [ ] An assignment from the inventor(s) of the patent application/patent identified above. The assignment was recorded in the United States Patent and Trademark Office at Reel _____, Frame _____, or for which a copy therefore is attached.

OR

B. [✓] A chain of title from the inventor(s), of the patent application/patent identified above, to the current assignee as follows:

1. From: Philip Rousselle To: Advent Networks, Inc.

   The document was recorded in the United States Patent and Trademark Office at
   Reel 014551, Frame 0422, or for which a copy thereof is attached.

2. From: Ronald Ingalls Trustee for Advent Networks, To: Inceptia LLC

   The document was recorded in the United States Patent and Trademark Office at
   Reel 019754, Frame 0232, or for which a copy thereof is attached.

3. From: _____ To: _____

   The document was recorded in the United States Patent and Trademark Office at
   Reel _____, Frame _____, or for which a copy thereof is attached.

[ ] Additional documents in the chain of title are listed on a supplemental sheet.

[✓] As required by 37 CFR 3.73(b)(1)(i), the documentary evidence of the chain of title from the original owner to the assignee was, or concurrently is being, submitted for recordation pursuant to 37 CFR 3.11.

[NOTE: A separate copy (*i.e.*, a true copy of the original assignment document(s)) must be submitted to Assignment Division in accordance with 37 CFR Part 3, to record the assignment in the records of the USPTO. See MPEP 302.08]

The undersigned (whose title is supplied below) is authorized to act on behalf of the assignee.

/Benzion A. Wachsman/ — Signature

September 26, 2008 — Date

Benzion A. Wachsman — Printed or Typed Name

718-928-2213 — Telephone Number

General Manager — Title